DOCUMENT RESUME

ED 151 022 IR 005 767

AUTHOR Romaniuk, E. W.; Montgomerie, T. C.
TITLE After Implementing Your CAI Course--What's Next.
INSTITUTION Alberta Univ., Edmonton. Div. of Educational Research Services.
REPORT NO DERS-05-031; RIR-76-2
PUB DATE Jun 76
NOTE 16p.; Paper presented to the Second Canadian Symposium on Instructional Technology (Quebec City, June 21-23, 1976) ; Not available in paper copy due to print quality of document

EDRS PRICE MF-$0.83 Plus Postage. HC Not Available from EDRS.
DESCRIPTORS *Computer Assisted Instruction; *Computer Oriented Programs; Course Evaluation; Higher Education; *Item Analysis; Man Machine Systems; *On Line Systems; *Performance; Programing; *Students

ABSTRACT

This paper follows the evaluation of a system of student performance analyses programs which has advanced from a crude manual system to an easy to use, on-line, interactive system. Recently, emphasis has focused upon the development of a system which requires little computer expertise on the part of the author to obtain a concise, easy to read summary of student performance at each node in the course. A CAI author can quickly examine the student performance listing and decide, based upon the student responses, which specific answer categories need to be added to the code and/or whether the instructional strategy should be modified or augmented. These changes can be noted on the performance listing by the author and passed to a programmer for course modification. The author needs only to wait until another group of students has taken the revised section of the course to compare changes in student performance on the revised version to that on the original version. (Author/VT)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED151022

DERS-06-031

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
NATIONAL INSTITUTE OF EDUCATION

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY.

BEST COPY AVAILABLE

RIR-76-2

AFTER IMPLEMENTING YOUR CAI COURSE-- WHAT'S NEXT

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

E.W. Romaniuk

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC) AND USERS OF THE ERIC SYSTEM."

Presented to: Second Canadian Symposium
on Instructional Technology
Quebec City, P.Q.
June 21-23, 1976

E.W. Romaniuk
T.C. Montgomerie

Division of Educational Research Services
Faculty of Education, The University of Alberta
Edmonton, Alberta T6G 2G5

IR000767

AFTER IMPLEMENTING YOUR CAI COURSE--WHAT'S NEXT

ABSTRACT

This paper follows the evolution of a system of performance analyses programs developed at the Division of Educational Research Services (DERS), at The University of Alberta. At DERS the analyses of student performance data has advanced from a crude manual system to an easy to use, on-line, interactive system. Most recently, emphasis has been focused upon the development of a system which requires little computer expertise (e.g. no knowledge of job control language is required) on the part of the author in order to obtain a concise, easy to read summary of student performance at each node in the course. Currently, a DERS CAI author who wishes to obtain a summary of student performance does so interactively at an on-line terminal. After being quizzed by the interactive system, regarding certain performance options desired by the author, the system automatically submits a batch request to an AMDAHL 470 computing system. The performance analyses system merges the course source code with the student performance data. The final result is a listing containing, for each node in the course, a copy of the CRT display and the frequency tabulation for all answer categories. In essence this means that an author receives an extremely concise hard copy output of student performance at each node in the course. In addition to the copy of the CRT display at each question, also included is a frequency matrix describing the number of student responses falling in the various answer categories on the students' first, second, third, fourth, and fifth (or greater) attempt. The author can also request that all student entries falling into certain answer categories be listed, along with each student's ID number.

The paper describes how a CAI author can quickly examine the student performance listing and decide, based upon the student responses, which specific answer categories need be added to the code and/or whether the instructional strategy should be modified or augmented. These changes can be noted on the performance listing by the author and passed to a programmer for course modification. The author needs only to wait until another group of students has taken the revised section of the course and then easily compare changes in student performance on the revised version to that of the original version. Because of the current availability of the performance analyses system, CAI authors at DERS are actively modifying their courses based on student performance rather than on intuition.

INTRODUCTION

The Division of Educational Research Services (DERS), Faculty of Education at the University of Alberta has operated an IBM 1500 Instructional Computer for the past eight years. Initially, emphasis of those working with the DERS CAI facility was upon building short courses. This was done in order to examine the possibilities of CAI as well as to examine the effects of different instructional strategies on student performance. At present, while short courses are still being used for research purposes, major emphasis has shifted towards the development of courses which range in length from 30 to 100 or more student contact hours and which may, in part or completely, replace conventionally administered courses.

One of the capabilities often touted by users of CAI is the ability to easily collect voluminous amounts of information about the student as that student progresses through a course. This information is referred to as "performance data" and includes the student's response to a question, the classification of that response by the computer, the time taken by the student to respond, and the current status of the student. When courses are of the short "research" variety, it is not too difficult, although often time consuming, to manually extract the specific information an author may require from the printed performance data. When courses reach the 30 to 100 hour range, attempts at manually extracting meaningful information from a voluminous performance data base, such as that for 100 or more students, can present an inconceivably arduous task.

Our experience in working with CAI authors is that those authors who produce courses of significant length, quickly realize that either they have to ignore most of the performance data, or that some automatic way of compiling the performance data into meaningful form must be found. To aid authors in the task of analyzing performance data, DERS undertook to find methods of sorting and compiling course performance data so that the author would be presented with a complete and meaningful summary of student performance.

This paper describes the development and operation of a system for merging student performance data with the original course code in order to give the author much of the information necessary to easily evaluate, modify and improve his course.

WHAT ARE PERFORMANCE DATA, AND WHAT CAN YOU DO WITH THEM?

Performance data are specific pieces of information regarding the current status of students. These data are

automatically recorded by the computer system at times specified by the course author. Two very important kinds of information can be examined.

1. The tracing of an individual student through the course in order to find points where the student is encountering difficulties. Depending on the data, the author can decide if human intervention is required.

2. The cumulative frequencies within response categories for all students at each question node in the course. This is done in order to identify areas in the course which may be in need of modification.

With this type of information, the author is well prepared both to assist a student who may have encountered particular difficulties, and to modify the course where a large proportion of the students appear to have encountered similar difficulties.

Since the DERS performance analyses programs were written specifically for data produced by the IBM 1500 Coursewriter II system, performance data will be discussed in light of this system.

In the Coursewriter II system, each question to be presented to the student is assigned a unique identifier (EPID) by the author. The student response is matched against a set of author specified anticipated answers. Each anticipated answer to each question is also assigned a unique "match" identifier (MAID). Each time a student is asked a question the system begins to collect data. Data accumulation for a particular question continues until the next question is presented; at this time, all data from the previous question is dumped to magnetic tape. The types of data dumped include:

1) the student ID

2) the course name

3) date and time

4) latency time (elapsed time from the time a question was presented until the response was received)

5) the question identifier (EPID)

6) the match identifier (MAID) which this

response matched

7) the actual student response

8) the contents of all numerical counters, binary switches, and registers.

Every record for each student is written in real time and is approximately 360 bytes long. Thus if the course contains many questions and has been accessed by a large number of students, a great deal of processing needs to be done before meaningful performance output can be returned to a course author.

Numerous attempts at providing the course author with meaningful student data have been made at other CAI Installations. One example is the "STRAP System" at the Ohio University (IBM, 1973). These programs, while suggesting possible approaches to the problem, were felt to be inadequate to meet the main criteria specified by the staff and authors working on the IBM 1500 system at DERS.

The next section of this paper traces the historical development of a set of the group performance data handling programs developed at DERS.

GROUP ANALYSES: VERSION 1 (Manual)

As more and more emphasis was placed on the production of full length courses, it quickly became apparent that while the analyses of the performance recordings of short courses could be undertaken by a manual system, the performance analyses of full length courses was becoming a logistic impossibility.

A second factor also became apparent. Authors who wrote short courses to be used for experimental purposes, most often desired to follow the paths of individual students through the complete course. With the switch to the writing of "credit courses," this emphasis shifted. Now authors wished to trace an individual student through a course only when the student appeared to be in difficulty or complained of incorrect course branching. Authors were more preoccupied with course improvement. They wished to know how all students were doing at each node point in a course. If points were found where students appeared to have had difficulty with the material, then the flow of each individual student through these points would be studied. This would enable authors to improve their courses so that future students would not encounter similar difficulties.

In order to find this information, authors had to:

1. Identify the EPID of each question under examination.

2. Page through a listing of the real time performance data tape, looking for each occurrence of that EPID, and manually collating student responses by MAID.

3. Examining the collated data to find similarity in student responses.

Needless to say, very few authors spent the enormous amount of time required for this task.

GROUP PERFORMANCE ANALYSES: VERSION 2

In 1970, a major attempt was made to automate the sorting of performance analysis data in order to give this information.

In order to use Version 2 of the performance analyses program, the author was required to punch, on computer cards, the EPID's and the associated MAID's for questions which he wished analysed. A short description of each question and the anticipated answers could be included. The analyses program first sorted the performance data for the desired particular course, or course segment, and then matched these sorted EPID's and MAID's against those punched on cards by the course author. For each question, frequencies for each answer category were then calculated and presented in a tabular format.

These methods of analyses in Version 2 certainly proved to be much easier to use than the manual system. However, Version 2 was still found to be very time consuming for the author. First, decisions had to be made regarding which question nodes were to be analyzed. Then the descriptive text for each question and associated anticipated answers had to be keypunched. Most authors claimed that it took them as long to design the performance analyses script as it did to design the course. Therefore, many authors failed to use the performance analyses programs.

GROUP PERFORMANCE ANALYSES: VERSION 3

By 1973, a strong need was felt for an improvement to the original group performance recording analyses program. Discussions with major authors showed that while they were happy with the statistics produced by Version 2 they found

the time required to use the programs quite prohibitive. It was decided that an attempt would be made to design programs which would use the original Coursewriter II course code to automatically produce card images equivalent to those which the author would have produced for Version II. Upon further study of the problem, it was realized that it would be possible to reproduce most of the actual content of the CRT screen at the time the student was asked each question. A table containing all anticipated correct, wrong, additional or unidentified answers could also be automatically produced. A program to implement these options was completed by January, 1974.

Version 3 worked as follows. First, a linear scan of the instructions in the original course code was made, producing a close approximation of the content of the CRT display each time a question was asked. Immediately following each question display, a list of the answer analyses for that question was produced, showing the various answers identified as either correct, wrong, additional, or unanticipated. In the case of special 1130 assembler functions used in answer analyses (e.g., key word, numerical limit, or Boolean relationships) strings of words or word roots were printed. The intermediate code was automatically stored on magnetic tape so that when subsequent students took the course, this part of the program would not have to be rerun.

When the students completed a segment (chapter) of the course, their performance data were sorted according to the question identifier (EPID) and sub-sorted according to date and time. The sorted performance data were then merged with the intermediate code giving a frequency count for each answer category within each question. An included option made it possible for the author to have the actual students' responses printed for each answer category. This option allowed authors to study those answers which had not been expected. Based upon these data the author could then modify the course, instructional material, add or delete certain answer categories, modify the question material, or modify the responses provided to the students. Examples of the final product of the sorted performance data merged with the intermediate code is shown in Figure 1.

Due to the relative ease of the use of this program, it was found to be much more readily accepted by the DERS CAI authors. As more authors began using the performance analyses program, a number of "suggested improvements" were made and a "final" improved version of the program was produced.

GROUP PERFORMANCE ANALYSES: VERSION 4

A common criticism of Version 3 of the performance analyses program was that, while an author could easily see how many times each answer category had been chosen, it was not possible to find out whether a particular answer category had been chosen on the student's first, second, or third or subsequent attempt. The program was modified to remedy this deficiency. Student responses were first sorted according to EPID, MAID, date, and time for each student. An output table was created which contained frequencies in each category according to the number of the attempt. A sample of the output is contained in Figure 1. As can be seen, the answer categories are listed horizontally while trials 1, 2, 3, 4, and more than four are listed in columns. An author can thus look at the analyses and see whether his prompts after wrong or unanticipated answers are leading the student towards the correct answer.

Another criticism of Version 3 was that users were either not aware of how to use some of the program options, or were not sure how to prepare the control language necessary to run the program. For example, errors in the control language often meant many false attempts before an error-free run was made. This criticism was overcome by the writing of an interactive program which explained the use of the options in the program, queried the user as to what options he wished to exercise, prepared the control commands to run the performance analyses programs, and automatically submitted the job to the remote job queue at the University of Alberta Computing Service's AMDAHL 470. An example of a typical run of the interactive program is given in Figure 2.

USE OF THE GROUP PERFORMANCE ANALYSES PROGRAM IN IMPROVING COMPUTER ASSISTED INSTRUCTION PROGRAMS

In the Coursewriter II authoring language the four available answer categories are correct, wrong, additional and unrecognizable. Authors at DERS have been encouraged to use the appropriate categories in their courses. For example, if a table of contents is to be presented to the student and choices are to be made that are neither correct nor incorrect, then the 'additional' answer category would be used. This is particularly important for those authors who wish a chapter summary, per student, of the total number of responses in each answer category.

Prior to having students take the course, a special backgrond utility program, called UNIQUE, is run against the source code to insure that the two character MAID associated with each answer is unique for each question and also that it reflects the answer categories (e.g.,

correct answer MAID'S begin with 'c1' wrong answer MAID'S with 'w1', etc.). This makes it easy for the author, or anyone else for that matter, who wishes to improve the course to glance at the summary performance analyses listing to check the answers provided and quickly spot with which category the answers are associated. Initially the author would most likely look for the nodes at which the sets of answers supplied by the author were inadequate. Often there can be large discrepancies between the responses anticipated by the author and those actually entered by the student. If this indeed is the case, the author could check the list of unanticipated answers and add specific answer analyses to diagnose and respond appropriately to groups of answers previously unanticipated. Occasionally, some of the unanticipated responses are either a result of a poor question stem or a misconception due to misleading information prior to the question. Another cause of multiple unanticipated student responses is the lack of appropriate feedback messages. Messages such as 'No', Wrong', 'Try again', 'Your answer does not compute', etc. do very little in guiding the student to the correct answer. If these are the only types of feedback messages provided to students when they enter unanticipated responses, it is no wonder that the list of this type of response often can be quite lengthy.

After checking the unanticipated responses, the author would next check to see what proportion of the responses matched the wrong answer sets as opposed to the correct answer sets. One might expect at least half the students to respond correctly on the first attempt. If the majority of students are providing wrong responses on their first three or four attempts then a change in instructional strategy may be warranted. As with the unanticipated answers, the author should check the feedback messages to see if they are appropriate.

One feature of the performance analyses program is that the author can request summary data for the specific passes through the course. That is, if first-pass summary data are requested, only data up to and including the first correct response for each question will be summarized. This option became necessary as authors permitted students to gain some control over their own movements through a course. Because of varying student needs, some students will review certain sections three or four times while others will not review at all and this can confound the summary data for the author. However, it should be noted that most authors at DERS still request the total summary data in addition to other-pass data. Examples of first pass analysis and total summary data for the same question are given in Figure 1.

Usually the authors note their desired changes on the summary performance analyses listing. When all the

requested changes have been noted, the authors can either make their own desired changes or pass the listing on to a DERS programmer assigned to the course; some of the authors design, code, and debug their own course material and require no programming assistance. Once the code has been updated, the program 'UNIQUE' is run against the source code and performance data from another group can be collected. When the next group finishes the course, or even a portion of the course, the authors can quickly obtain a summary performance recording listing. Comparisons between the previous and current data can be made to see if the course modifications have produced the desired changes in student performance.

Finally, the performance analyses program provides the authors with three other types of relevant data:

a) A summary of all student comments are listed by label sequence-by-student. The comments are extremely valuable to the author in that students often provide instances of problems ranging from spelling errors to inappropriate answer analysis.

b) The total elapsed times are listed for each student in each course segment (chapter). These data are very important in helping the students schedule their CAI sessions.

c) A "proctor key" is optionally produced for the author. The set of all answers for each question and the associated EPID is listed. A sorted table of EPID's is also available. The proctor keys are available in the terminal room. In some courses, those students encountering difficulties at times when the academic staff member or graduate assistant conducting the course is not available, may request the current EPID from the system, get the proctor key, find the EPID in the EPID table, and then check the page referenced in the table to find the correct answer.

CONCLUSIONS

The author designing CAI course material needs access to a concise easily readable summary of student performance, if the course is to be optimized. One of the ways of quickly providing this material for the author is through an easy to use on-line, interactive set of programs. At DERS, an author needs to spend only five minutes, or less, at an interactive terminal to enter the

parameters required to run the performance summary programs (Montgomerie and Romaniuk, 1976). Because authors at DERS now spend little of their time in obtaining summary performance data, they are now spending much more time in revising and optimizing their courses. One of the phrases being heard less often is one such as the following:

> "I don't need to collect summary performance data because I know how well the students are performing; therefore, there is no real need for me to modify my code."

Actually much work remains after initially defining your course. Experience has shown that a course tends to become static only after three to five full modifications to the course, based on performance recordings. It is only then that less emphasis can be placed on performance recordings.

REFERENCES

IBM. IBM 1500 Coursewriter II, Author's Guide, (GY26-1580-0), San Jose, California, Author, 1968.

IBM. Student Response Analysis Processors (STRAP), Installed Users Program General Documentation, (G320-1441-0), Author, 1973.

Montgomerie, T.C. & Romaniuk, E.W. Interactive Programs: A Simple Way to Allow Naive Users to Manipulate Extensive Data Bases. Proceedings of the University of Alberta Users Applications Symposium, University of Alberta, May, 1976.

Webster, D. (Editor). MTS Command Language, Reference Manual Computing Services, The University of Alberta, Edmonton, September, 1974.

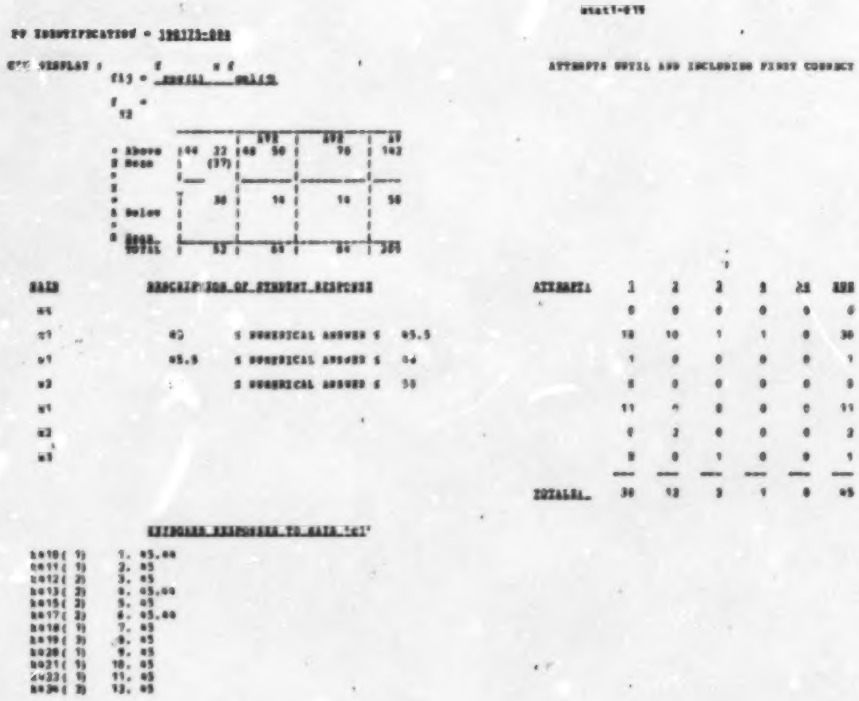

Figure 1. Output of Group Analyses Program.

stat1-019

```
. RP IDENTIFICATION = chisq7-028

CRT DISPLAY : In our original problem of tossing 300
              dice, we calculated the frequencies in
              the categories 1 through 6. What are the
                                                  2
              degrees of freedom (df) for X  ?
                                           2
              What is the critical   X    =
                                    .05 (5)
                                         2
              Assume that:        obs(5) =  8.72
              WOULD YOU ACCEPT OR REJECT THE NULL
              HYPOTHESIS OF NO DIFFERENCE BETWEEN THE
              THE OBSERVED AND EXPECTED FREQUENCIES?
              AAAAPP              BBBBBV
```

RAID	DESCRIPTION OF STUDENT RESPONSE
a0	
a1	AAAAPP
b1	BBBBBV
u1	

ATTEMPTS UNTIL AND INCLUDING FIRST CORRECT

ATTEMPT:	1	2	3	4	25	RRR
	0	0	0	0	0	0
	25	5	0	0	0	30
	3	0	0	0	0	3
	3	1	1	0	0	5
TOTALS:	31	6	1	0	0	38

LIGHTER (ROW-COL) AT RAID 'a1'

```
R020( 1)    1.  25 ,  13
R020( 2)    2.  26 ,  10
R020( 3)    3.  26 ,  13
R030( 1)    4.  28 ,   2
R033( 1)    5.  26 ,   3
```

stat1-019

```
. RP IDENTIFICATION = chisq7-028

CRT DISPLAY : In our original problem of tossing 300
              dice, we calculated the frequencies in
              the categories 1 through 6. What are the
                                                  2
              degrees of freedom (df) for X  ?
                                           2
              What is the critical   X    =
                                    .05 (5)
                                         2
              Assume that:        obs(5) =  8.72
              WOULD YOU ACCEPT OR REJECT THE NULL
              HYPOTHESIS OF NO DIFFERENCE BETWEEN THE
              THE OBSERVED AND EXPECTED FREQUENCIES?
              AAAAPP              BBBBBV
```

RAID	DESCRIPTION OF STUDENT RESPONSE
a0	
a1	AAAAPP
b1	BBBBBV
u1	

ATTEMPT:	1	2	3	4	25	RRR
	0	0	0	0	0	0
	25	10	4	1	0	40
	3	0	1	0	0	4
	3	1	1	0	0	5
TOTALS:	31	11	6	1	0	57

LIGHTER (ROW-COL) AT RAID 'a1'

```
R020( 1)    1.  25 ,  13
R020( 2)    2.  26 ,  10
R020( 3)    3.  26 ,  13
R030( 1)    4.  28 ,   2
R033( 1)    5.  26 ,   3
```

Figure 1. Output of Group Analyses Program (Cont'd).

PROGRAM PERFORMGROUP

Do you understand the difference between Part 1 and Part 2 of Program PERFORMGROUP? YES

Do you want to run Part 1 or Part 2. (Type 1 or 2.) 1

Do you have the CWII COURSE LISTING TAPE FORM which was returned with your 1500 Service Request Form? YES

Please enter your Sign-On ID: CSID
Please enter your password:
Do you wish this output returned to Education? YES
What is your Bin number? 22
How many copies of the output do you wish? 2
What priority do you wish? L

On the CWII COURSE LISTING TAPE FORM there is a "VOLUME SERIAL" beginning with ER.
Please fill in the four digits - ER 6000

Is there more than one file on the tape? NO

What is the 5 character name of your course? COURS

What is the three digit segment number? (Please put in all leading zeros, eg. 000, OR 020). 000

Would you like a set of ep's and expected answers for the Proctor? YES

Do you wish all correct responses printed? YES

Do you wish all wrong responses printed? YES

Do you wish all additional responses printed? YES

Do you wish all unanticipated responses printed? YES

Do you wish all overtime responses printed? YES

Do you wish lightpen responses printed? YES

You may have all student responses tabulated, or only the responses for the first pass of each student through each section. Do you want ALL student responses tabulated? YES

Do you wish to use any special character for over-striking alternate coded characters? (The default is a "bullet".) NO

Figure 2. Sample Interactive Session

```
*****  THIS FILE HAS BEEN CALLED  *****
              PRCODE.000.COURS
*****  PLEASE MAKE A NOTE OF THE NUMBER  *****

Are you ready to run Part 2 now? YES

Before you run Part 2 on any set of performance data,
you MUST have had that data separated into segments.
Has this been done yet? NO

Before you go any further, you must arrange for a tape
which will contain your performance data sorted into
indvidual segments.  This will be referred to as your
"SEGMENTS" tape. Have you arranged for this tape yet?  YES

Let's look first at the "AUTHOR" tape
  What is the rack number? EO  6666
  Now the "VOLUME SERIAL"  ER  0213

Four options for merging files are available:
  1) All files on the input tape may be merged.
  2) Selected files on the input tape may be merged.
  3) All files on the input tape, beginning at a
     specified file, may be merged.
  4) Files "M" to "N" inclusive on the input tape
     may be merged.
Type in the number of the option you wish. 1

Now I need to know about your "SEGMENTS" tape.
  What is the Computing Services Rack Number? EO  6123
  What is the "Volume Serial"? ER  0765

What is the five character name of the first course for
which Performance Records are stored on this tape?  COURS

Now type in ALL the segments of this course for which
Performance Records exist.  Type the segments in one at a
time. All leading zeros must be typed (eg. 021).
Type END when all segments have been entered.
  Enter a segment  000
  Enter a segment  END

Do you wish to recover the Performance Records for any
other courses from this tape? NO

Do you want to run Part 2 against all the segments
you ran Part 1 against today?  YES

Do you wish to run Part 2 against any segments which
have been run on Part 1 previously? NO

Will you be using the course listing tape for other
jobs? NO
```

Figure 2. Sample Interactive Session (cont'd.).